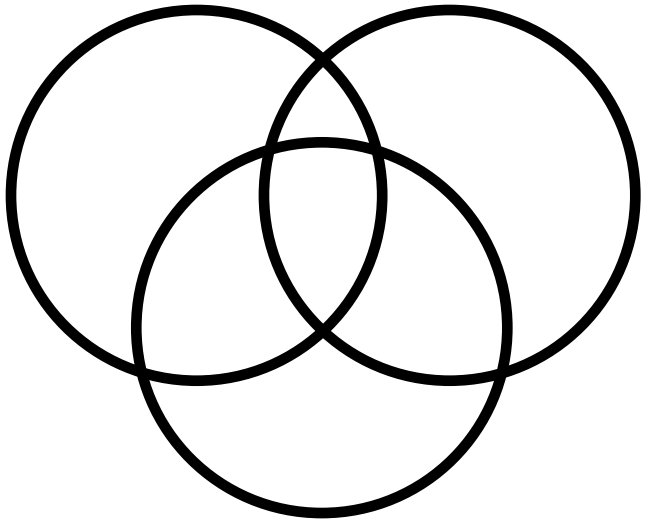
Security Models: Past, Present and Future

Prof. Ravi Sandhu
Executive Director and Endowed Chair
Institute for Cyber Security
University of Texas at San Antonio
August 2010

ravi.sandhu@utsa.edu
www.profsandhu.com

Security Objectives

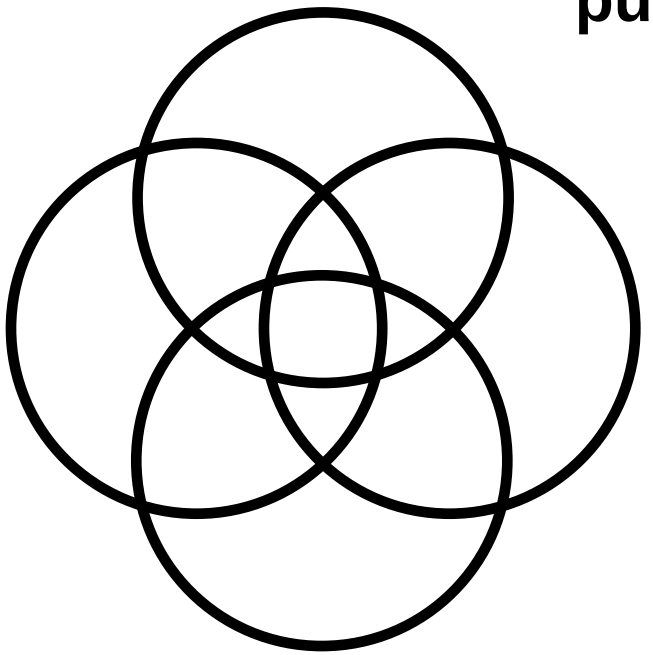


INTEGRITY
modification

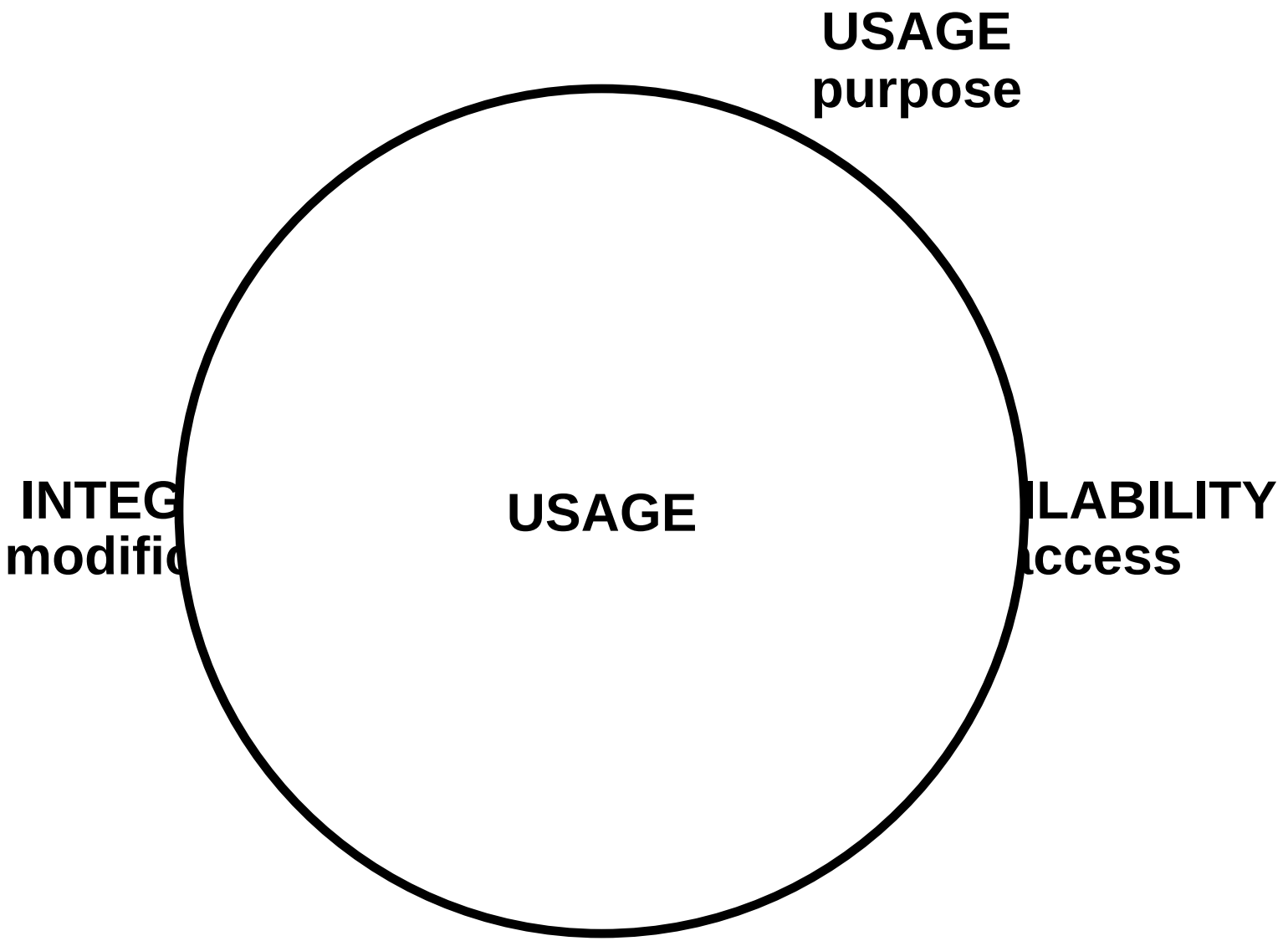
AVAILABILITY
access

CONFIDENTIALITY
disclosure

Security Objectives



Security Objectives



Butler Lampson Paraphrased (I think)

- Computer scientists could never have designed the web because they would have tried to make it work.
  But the Web does “work.”
  What does it mean for the Web to “work”?
- Security geeks could never have designed the ATM network because they would have tried to make it secure.
  But the ATM network is “secure.
  What does it mean for the ATM network to be “secure”?

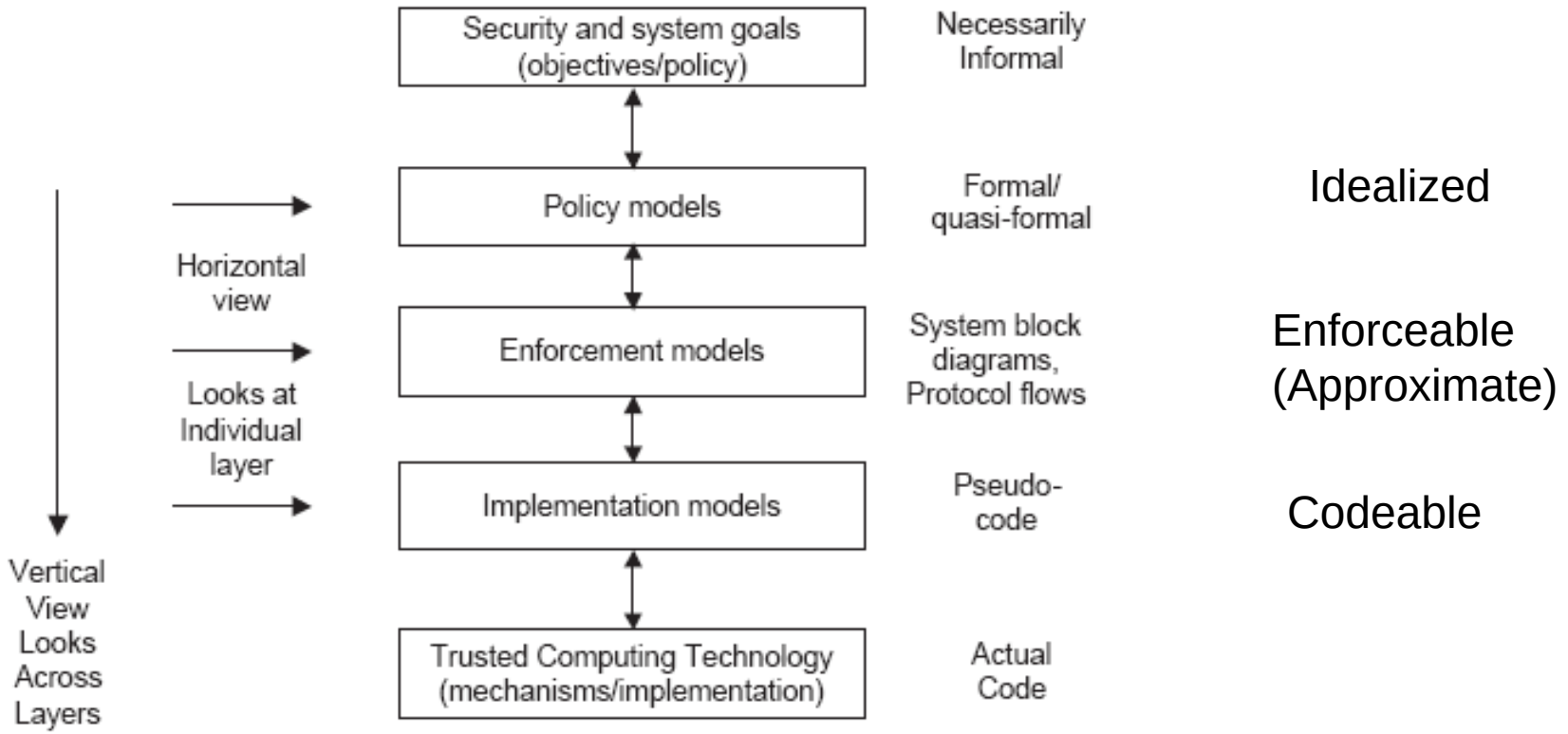
Foundational Security Assumptions

- Information needs to be protected
  - In motion
  - At rest
  - In use
- Absolute security is impossible and unnecessary
  - Trying to approximate absolute security is a bad strategy
  - "Good enough" security is feasible and meaningful
  - Better than "good enough" is bad
- Security is meaningless without application context
  - Cannot know we have "good enough" without this context
- Models and abstractions are all important
  - Without a conceptual framework it is hard to separate "what needs to be done" from "how we do it"

We are not very good at doing any of this

PEI Models: 3 Layers/5 Layers

This lecture is focused on the policy models layer



At the policy layer security models are essentially access control models

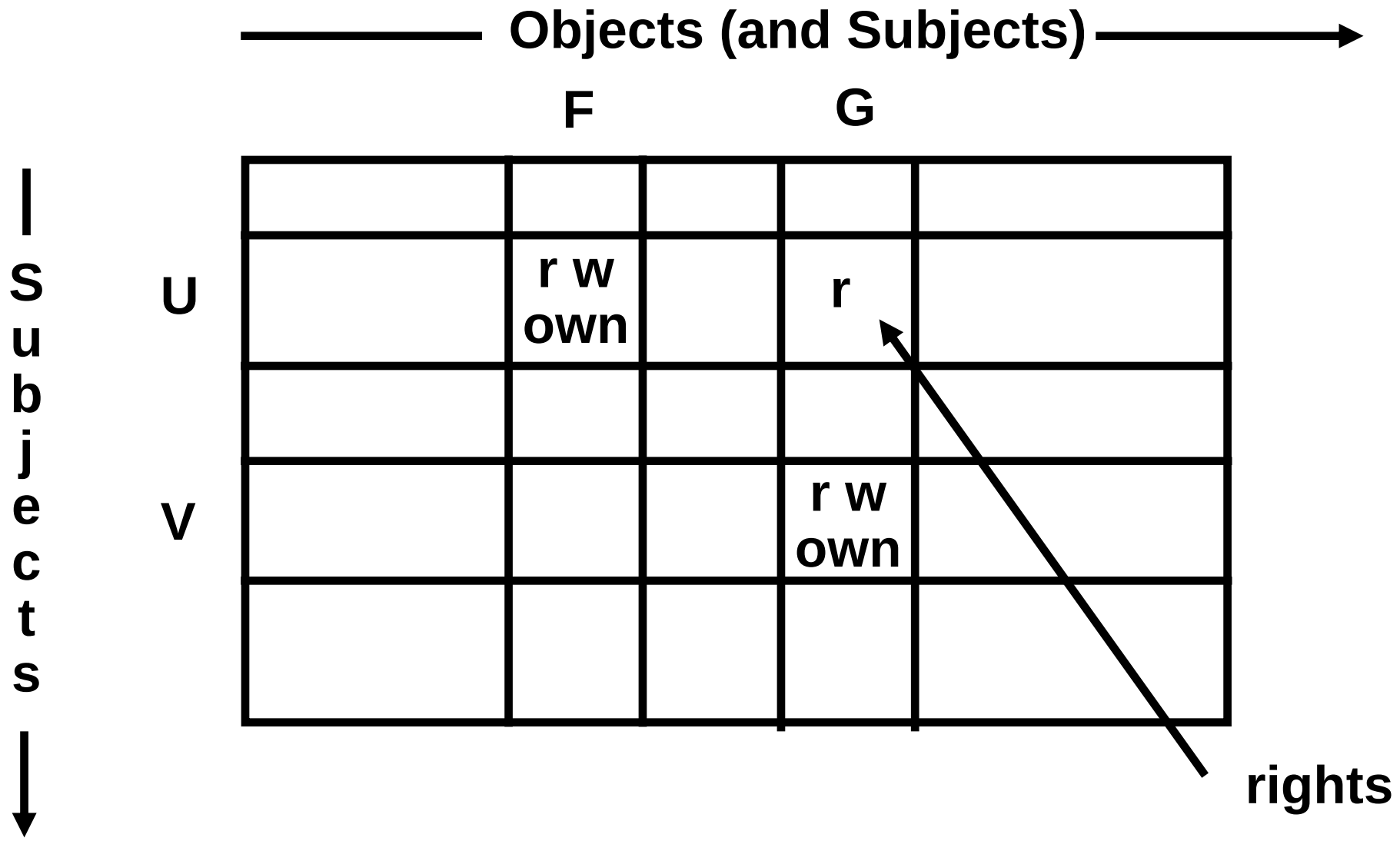
THE PAST

Access Control Models

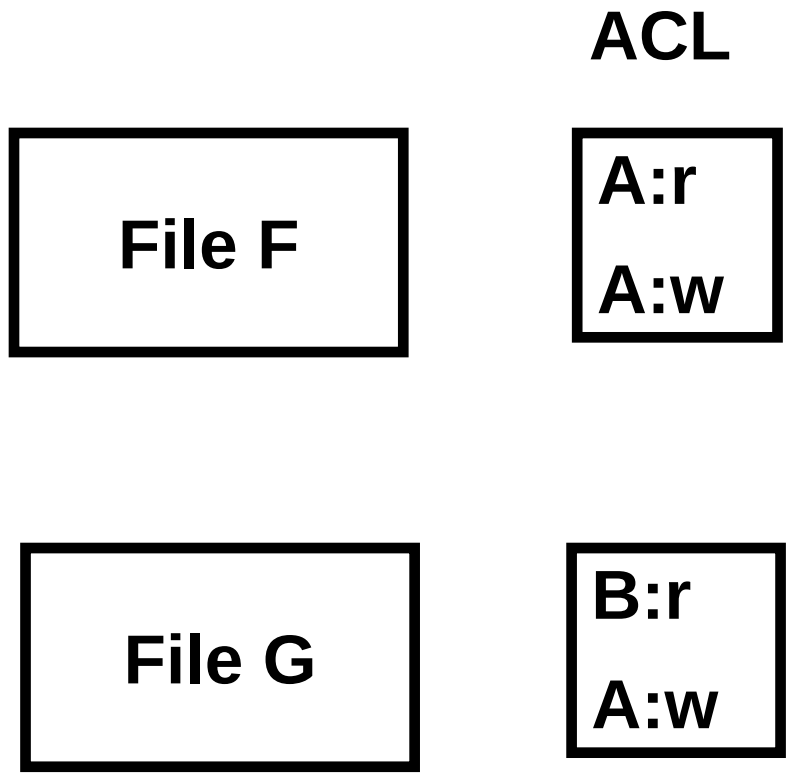
- Discretionary Access Control (DAC)
  - Owner controls access but only to the original, not to copies
- Mandatory Access Control (MAC)
  Same as Lattice-Based Access Control (LBAC)
  - Access based on security labels
  - Labels propagate to copies
- Role-Based Access Control (RBAC)
  - Access based on roles
  - Can be configured to do DAC or MAC
  - Generalizes to Attribute-Based Access Control (ABAC)

Numerous other models but only 3 successes

DAC: ACCESS MATRIX MODEL

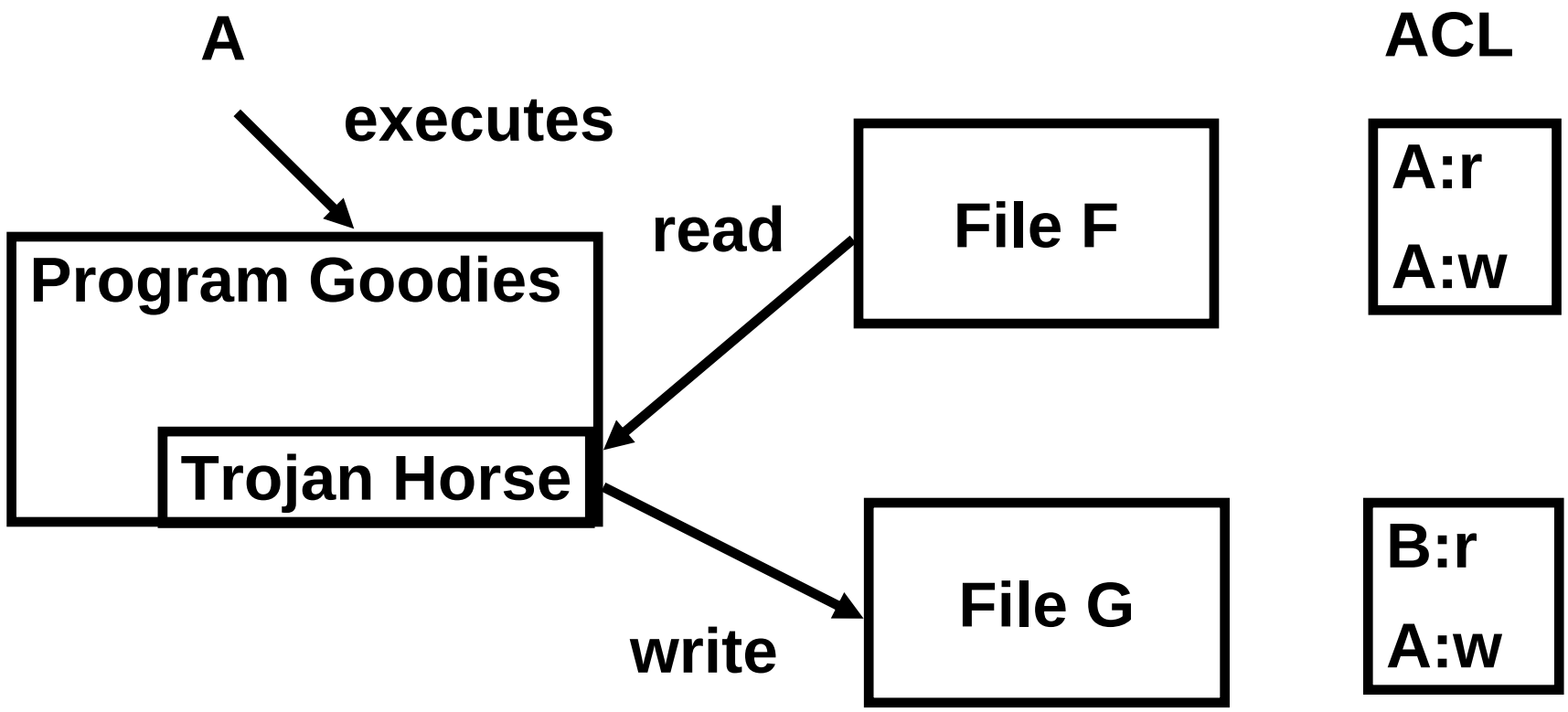


DAC: TROJAN HORSE EXAMPLE



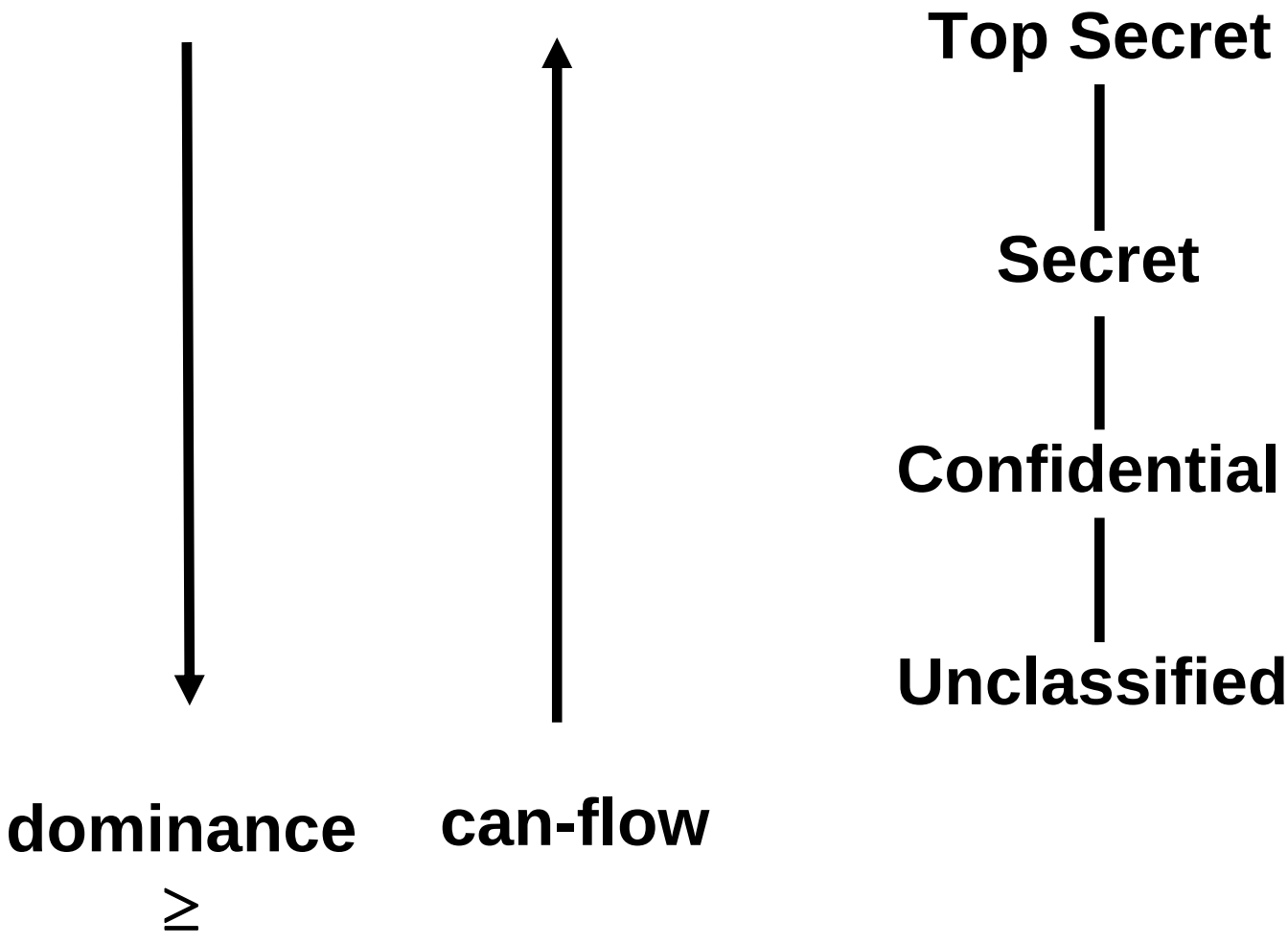
B cannot read file F

DAC: TROJAN HORSE EXAMPLE

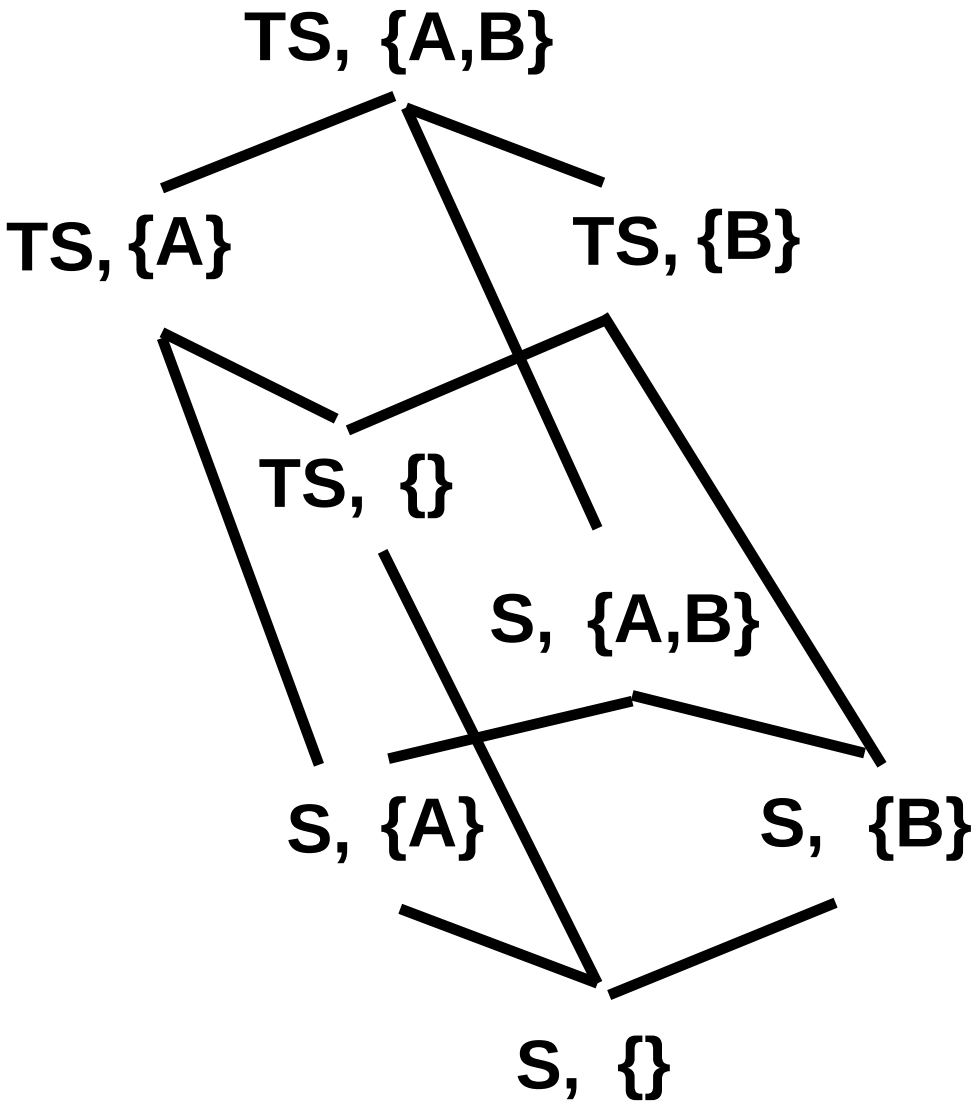


B can read contents of file F copied to file G

LBAC: LATTICE STRUCTURES



LBAC: LATTICE STRUCTURES



Hierarchical Classes with Compartments

LBAC: BELL LAPADULA (BLP)

SIMPLE-SECURITY

Subject S can read object O only if

- label(S) dominates label(O)

STAR-PROPERTY (LIBERAL)

Subject S can write object O only if

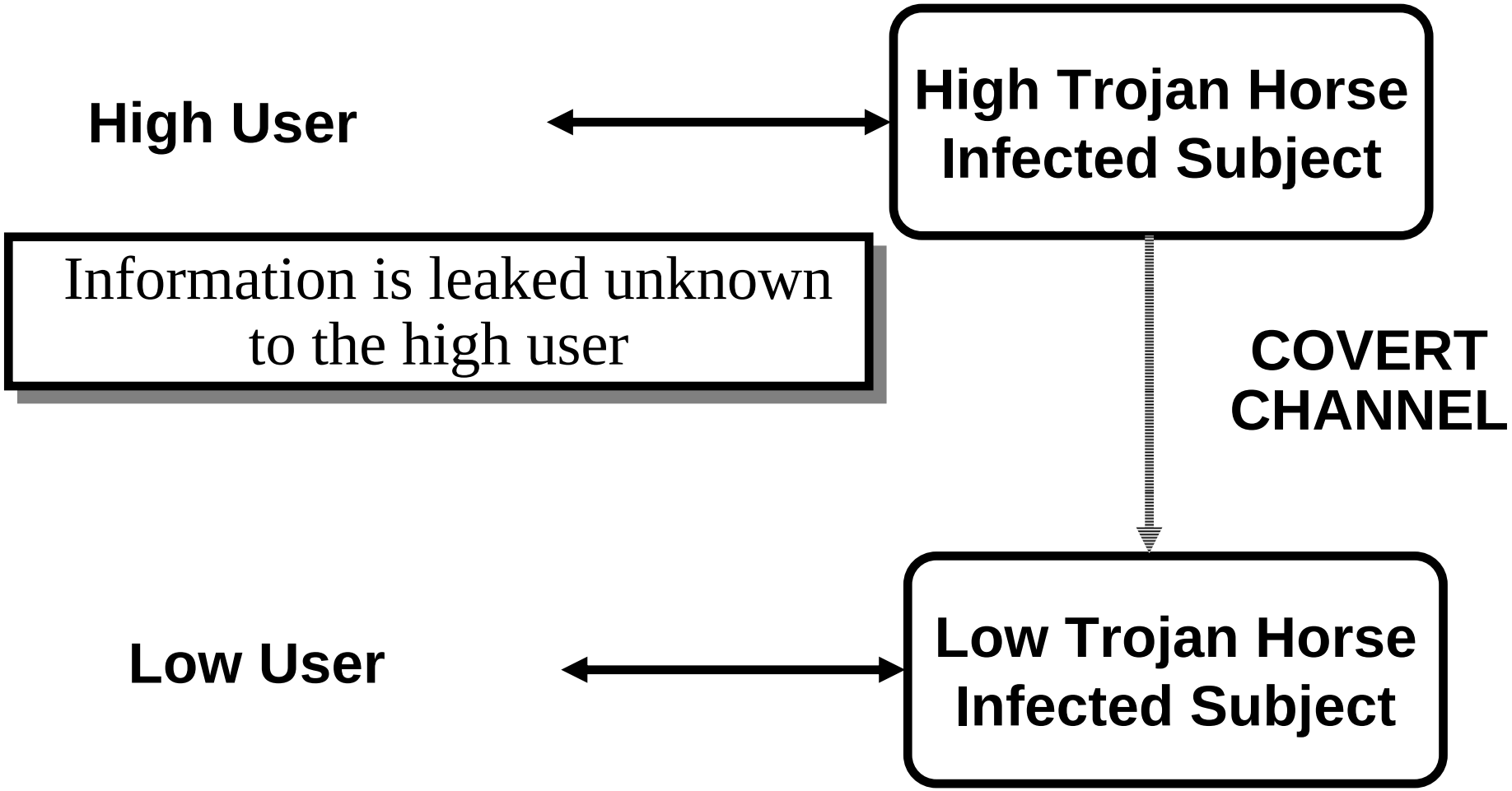
- label(O) dominates label(S)

STAR-PROPERTY (STRICT)

Subject S can write object O only if

- label(O) equals label(S)

LBAC: COVERT CHANNELS



RBAC: Role-Based Access Control

- Access is determined by roles
- A user's roles are assigned by security administrators
- A role's permissions are assigned by security administrators

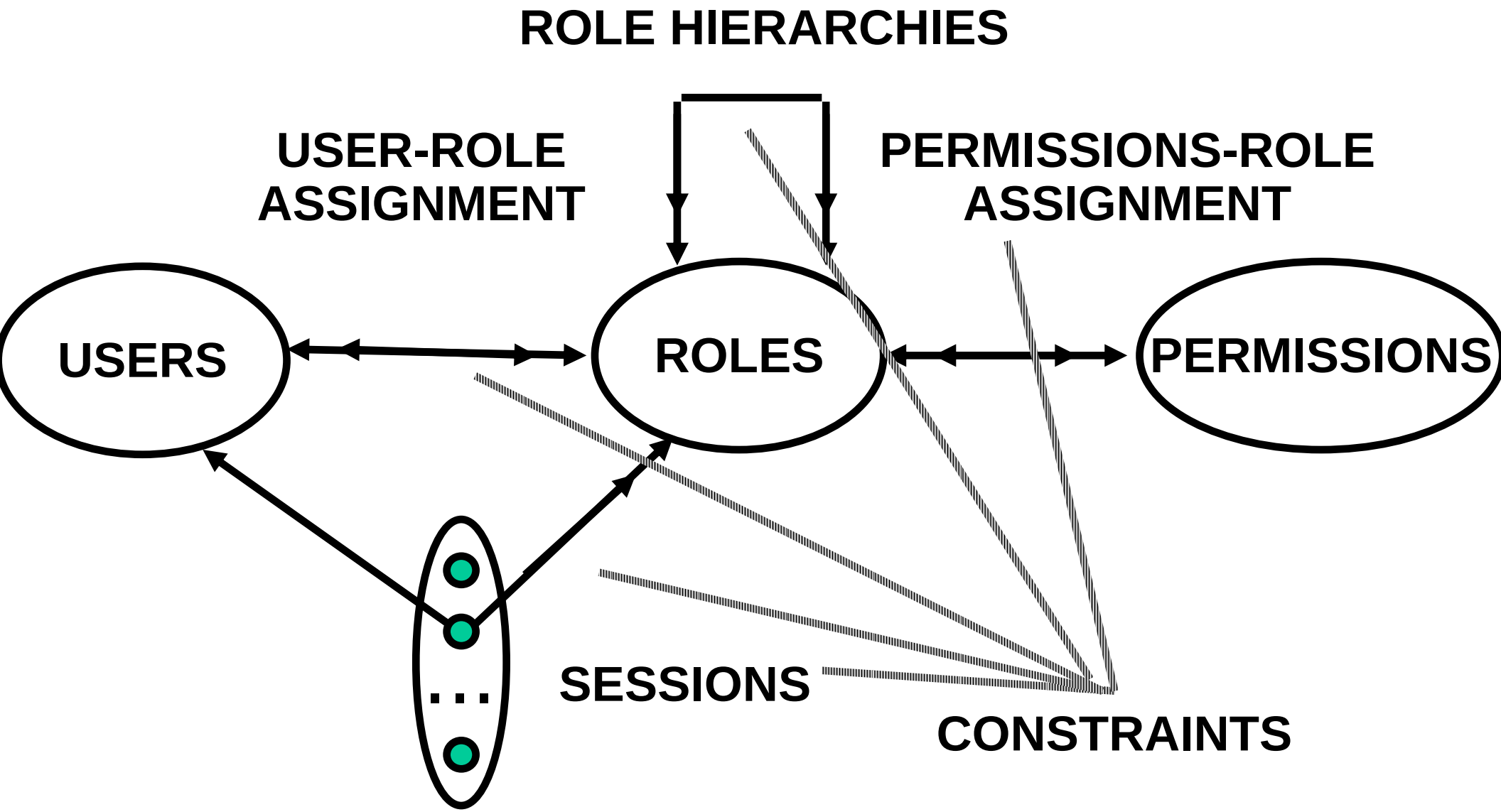
First emerged: mid 1970s
First models: mid 1990s

Is RBAC MAC or DAC or neither?

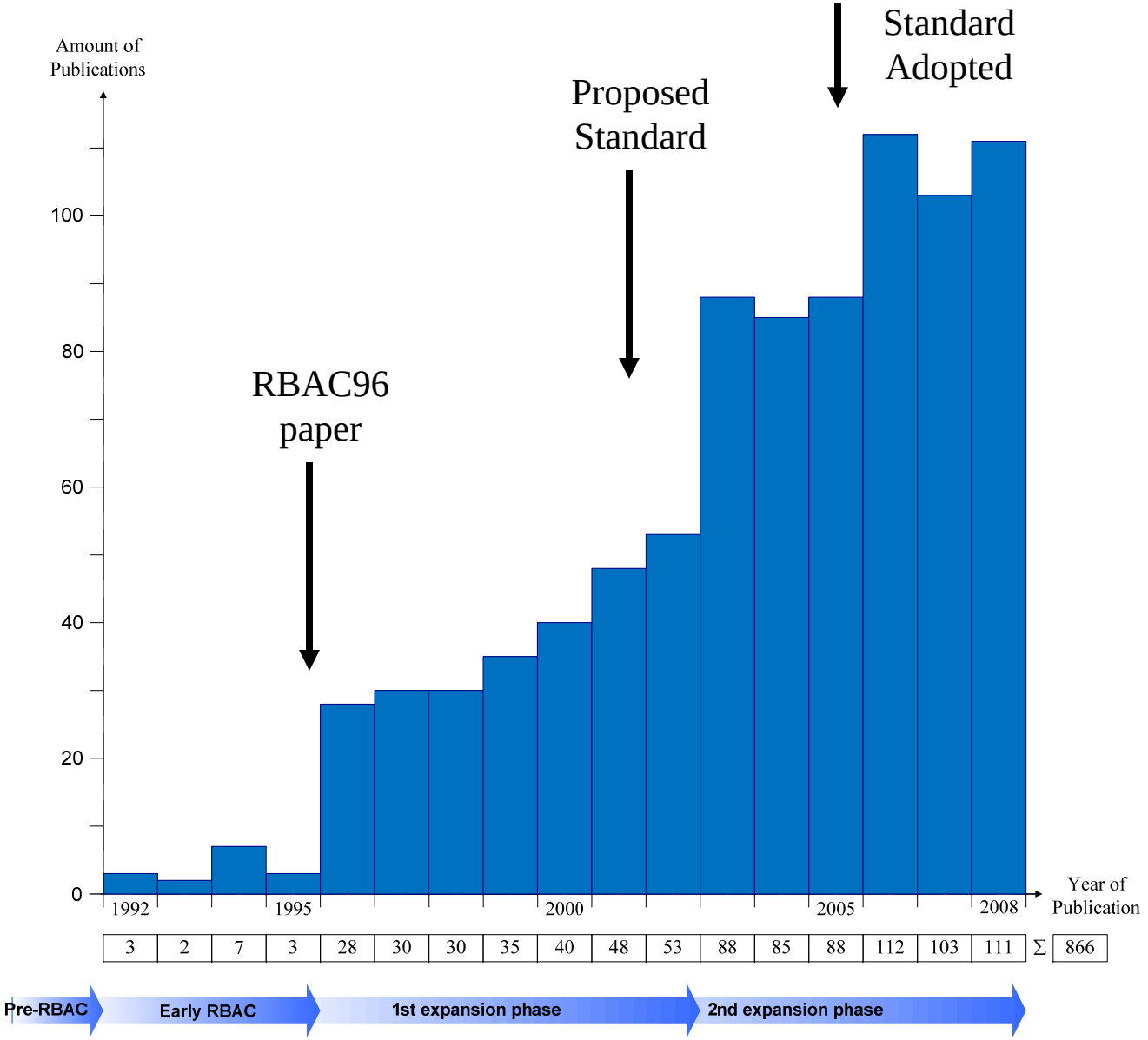
- RBAC can be configured to do MAC
- RBAC can be configured to do DAC
- RBAC is policy neutral

RBAC is neither MAC nor DAC!

RBAC: RBAC96 Model

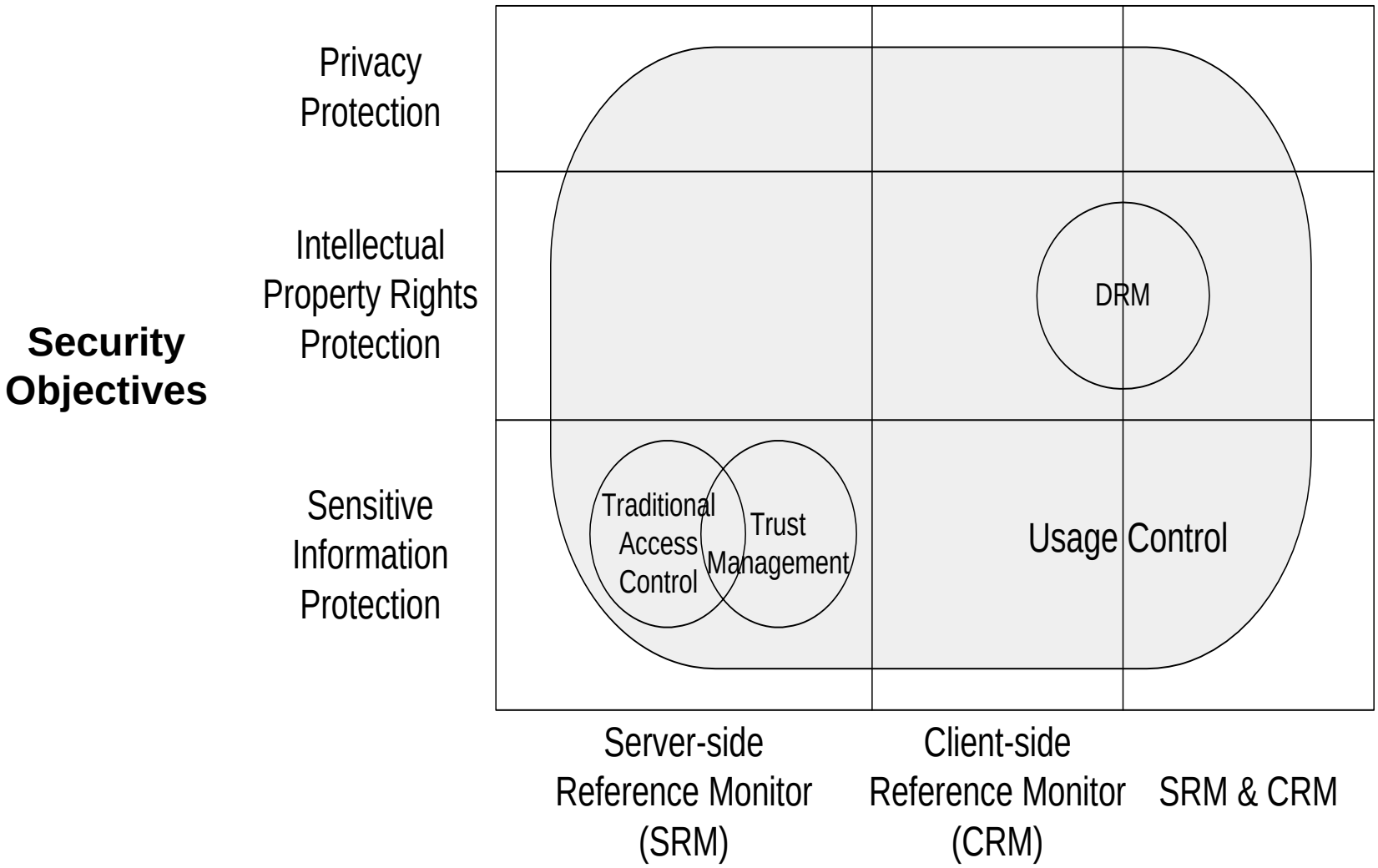


RBAC: The RBAC Story



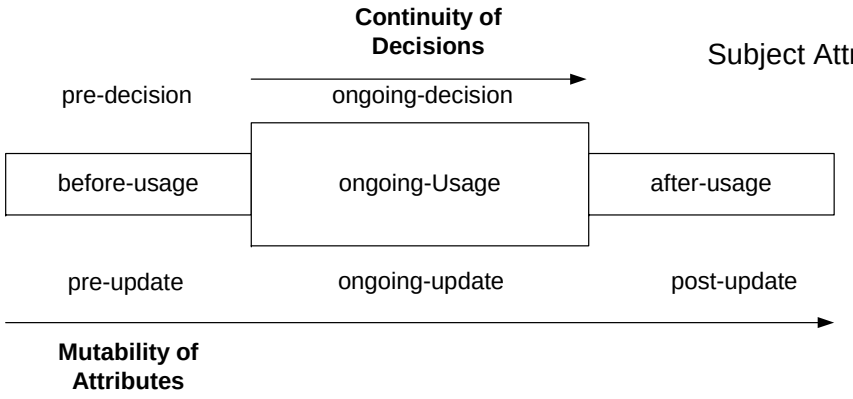
THE PRESENT

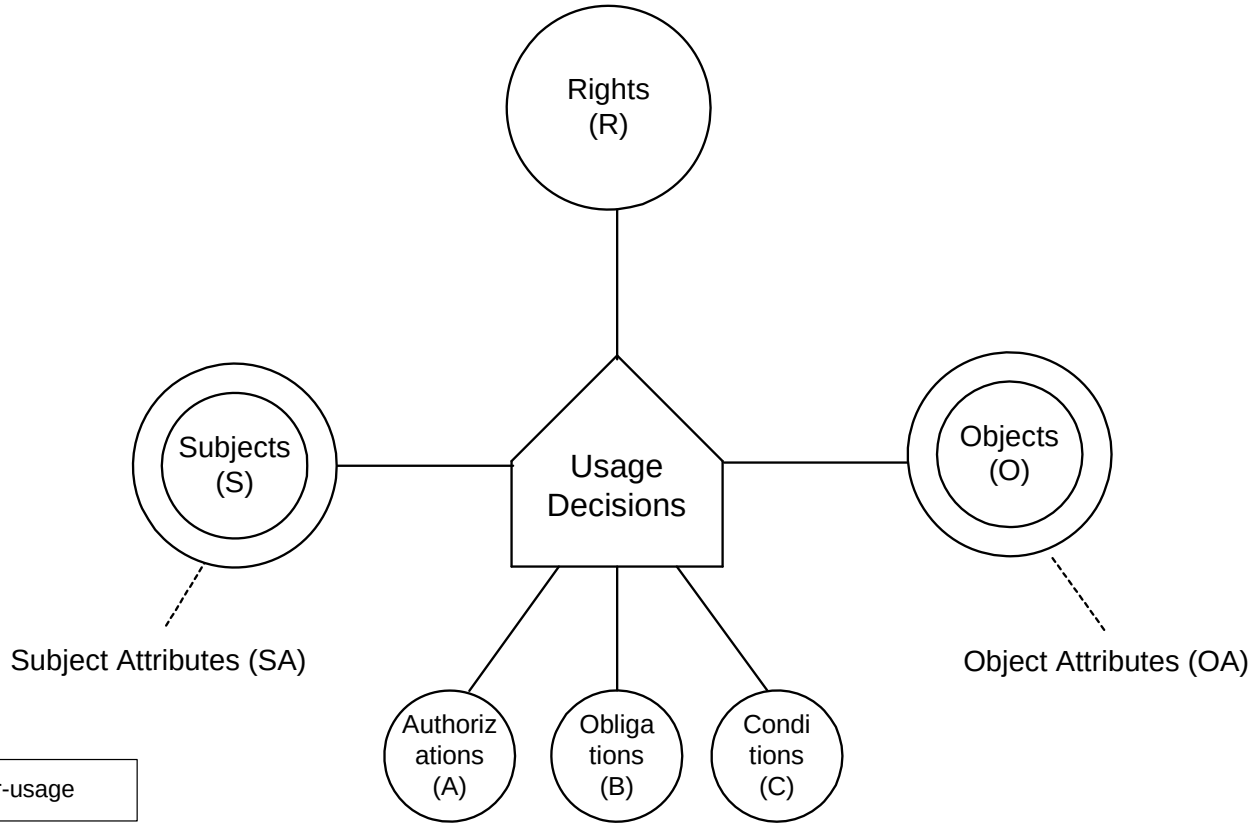
UCON: Usage Control Scope



UCON: Usage Control Model

- unified model integrating
  - authorization
  - obligation
  - conditions
- and incorporating
  - continuity of decisions
  - mutability of attributes





UCON is ABAC on steroids

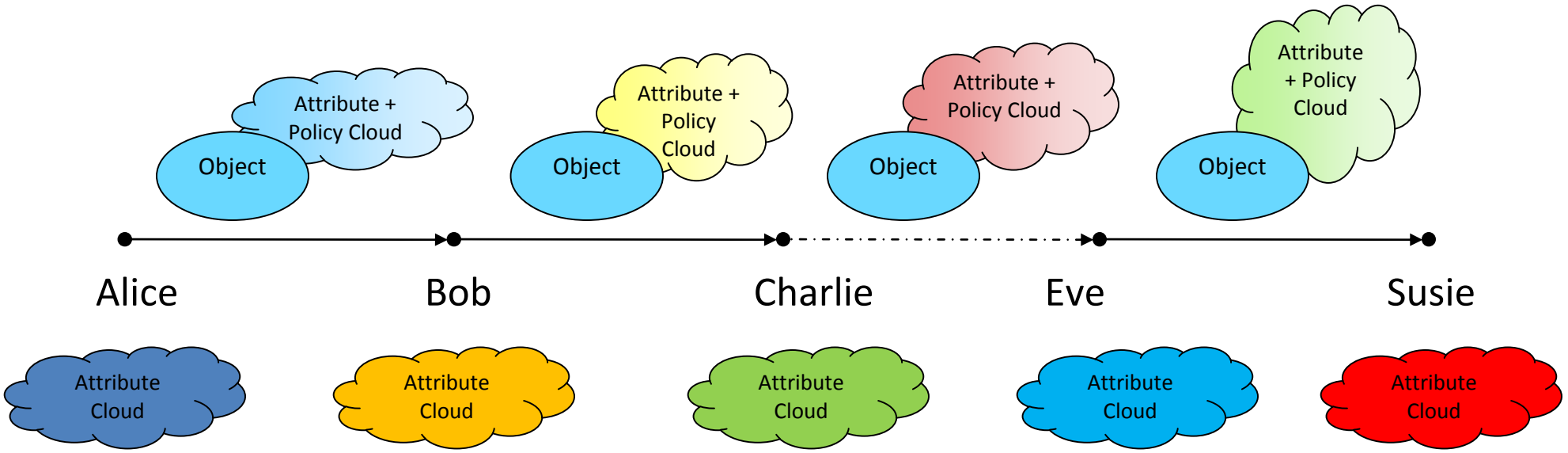
THE FUTURE

Application-Centric Security Models

- Our Basic Premise
  - There can be no security model without application context
- So how does one customize an application-centric security model?
  - Meaningfully combine the essential insights of
    - DAC, LBAC, RBAC, ABAC, UCON, etcetera
  - Directly address the application-specific trade-offs
    - Within the security objectives of confidentiality, integrity and availability
    - Across security, performance, cost and usability objectives
  - Separate the real-world concerns of
    - practical distributed systems and ensuing staleness and approximations (enforcement layer) from
    - policy concerns in a idealized environment (policy layer)

Dissemination-Centric Sharing

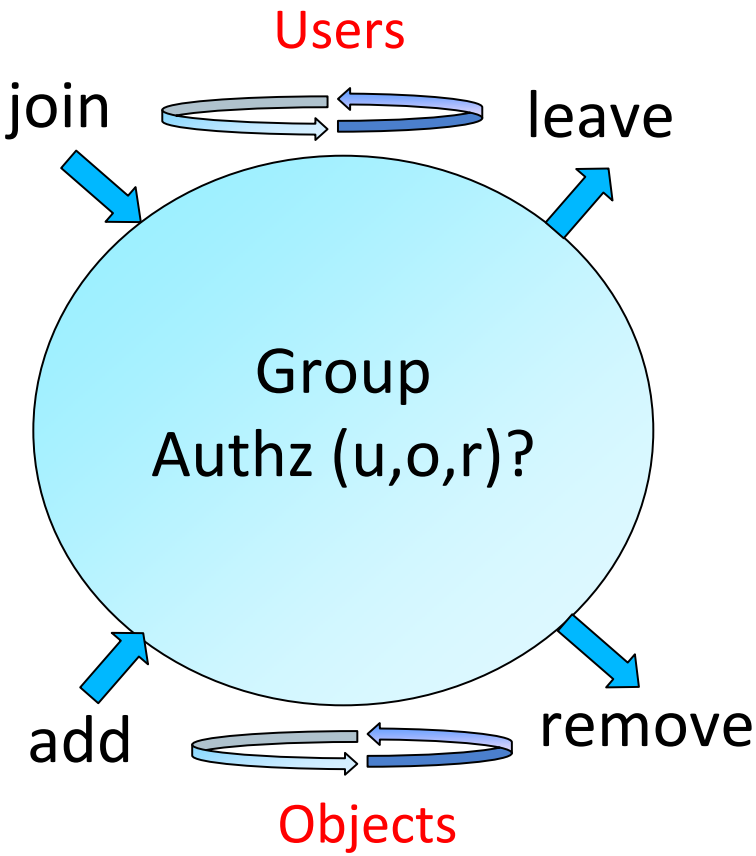
- Extensive research in the last two decades
  - ORCON, DRM, ERM, XrML, ODRL, etc.
- Copy/usage control has received major attention
- Manageability problem largely unaddressed



Dissemination Chain with Sticky Policies on Objects

Group-Centric Sharing (g-SIS)

- Brings users & objects together in a group
  - Focuses on manageability using groups
  - Co-exists with dissemination-centric
  - Two metaphors
    - Secure Meeting Room (E.g. Program committee)
    - Subscription Model (E.g. Secure multicast)
- Operational aspects
  - Group characteristics
    - E.g. Are there any core properties?
  - Group operation semantics
    - E.g. What is authorized by join, add, etc.?
  - Read-only Vs Read-Write
- Administrative aspects
  - E.g. Who authorizes join, add, etc.?
  - May be application dependant
- Multiple groups
  - Inter-group relationship



CONCLUSION

Conclusion

THE PAST

- Discretionary Access Control (DAC)
- Mandatory Access Control (MAC)
  - Equivalently Lattice-Based Access Control (LBAC)
- Role-Based Access Control (RBAC)

THE PRESENT

- Usage Control (UCON)
  - Attribute-Based Access Control (ABAC) on steroids

THE FUTURE

- Application-Centric Access Control Models
- Technology-Centric Access Control Models

Models are all important
A Policy Language is not a substitute for a good model
Lots of interesting/impactful research to be done at P, E and I layers